## Protokoll Factory Design Pattern

1) importieren der modules

**import** os  
**import** pygame  
**from** mutagen.id3 **import** ID3  
**from** tkinter **import** \*  
**from** tkinter.filedialog **import** askdirectory

2) Tkinter frame erstellen und größe festlegen

root= Tk()  
root.minsize(600,400)

3) Listen für die Lieder erstellen + Label um den derzeitigen Song anzuzeigen

listofsongs =[]  
realnames =[]  
v= StringVar()  
songlabel = Label(root,textvariable=v,width=35 )

4)Methoden um den nächsten,vorherigen Song auszuwählen und die Musik zu stoppen

**def** nextsong(event):  
 **global** index  
 index +=1  
 pygame.mixer.music.load(listofsongs[index])  
 pygame.mixer.music.play()  
 updatelabel()  
**def** prevsong(event):  
 **global** index  
 index -= 1  
 pygame.mixer.music.load(listofsongs[index])  
 pygame.mixer.music.play()  
 updatelabel()  
  
**def** stopsong(event):  
 pygame.mixer.music.stop()  
 v.set(**""**)  
 **return** songname

5) Methode um das Songlabel zu aktualisieren

**def** updatelabel():  
 **global** index  
 **global** songname  
 v.set(realnames[index])

6)Wählt den Ordner aus und überprüft ob es sich um ein Musikstück handelt, falls ja spielt es das File ab

**def** directorychooser():  
 directory = askdirectory()  
 os.chdir(directory)  
 **for** files **in** os.listdir(directory):  
 **if** files.endswith(**".mp3"**):  
 realdir = os.path.realpath(files)  
 audio = ID3(realdir)  
 realnames.append(audio[**'TIT2'**].text[0])  
  
 listofsongs.append(files)  
  
 print(files)  
  
 pygame.mixer.init()  
 pygame.mixer.music.load(listofsongs[0])  
 pygame.mixer.music.play()

7) Die Buttons ins Frame einbinden

nextbutton = Button(root,text =**'Next Song'**)  
nextbutton.pack()  
  
previousbutton= Button(root,text =**'Previous Song'**)  
previousbutton.pack()  
  
stopbutton = Button(root,text =**'Stop Music'**)  
stopbutton.pack()  
  
  
nextbutton.bind(**"<Button-1>"**,nextsong)  
  
previousbutton.bind(**"<Button-1>"**,prevsong)  
  
stopbutton.bind(**"<Button-1>"**,stopsong)  
songlabel.pack()